

# Implementasi Algoritma DBSCAN Dalam Mengelompokkan Data Pasien Terdiagnosa Penyakit Ginjal Kronis (PGK)

**Richardo Anggara<sup>1</sup>, Abdul Rahman, S.Si., M.T.I.<sup>2\*</sup>**

<sup>1,2</sup>Universitas Multi Data Palembang; Jl. Rajawali No. 14, Telp : 0711-376400

<sup>1,2</sup>Program Studi Informatika, FIKR, Universitas Multi Data Palembang

[richardoang@mhs.mdp.ac.id](mailto:richardoang@mhs.mdp.ac.id), [arahman@mdp.ac.id](mailto:arahman@mdp.ac.id)

## **Abstrak**

Penyakit ginjal kronis (PGK) adalah penyakit ginjal yang ditandai dengan kerusakan struktural atau fungsional pada ginjal yang berlangsung lebih dari 3 bulan. PGK ditandai dengan satu atau lebih tanda kerusakan ginjal, yaitu albuminuria, sedimen urin abnormal, elektrolit, histologi, struktur ginjal, atau riwayat transplantasi ginjal, dengan penurunan laju filtrasi glomerulus. Pada penelitian ini dilakukan implementasi algoritma DBSCAN untuk mengelompokkan data pasien terdiagnosis PGK. Basis data yang digunakan adalah Chronic Kidney Disease yang berjumlah 400 data medis dengan 24 atribut atau fitur. Hasil penelitian dievaluasi dengan mencatat banyak jumlah cluster dan noise yang didapatkan menggunakan metric euclidean. Hasil terbaik ada pada skenario kedua dengan nilai epsilon sebesar 3,5, dan  $Min\ sample=5$  yang menghasilkan jumlah cluster sebanyak 2 cluster dengan nilai Silhouette sebesar 0.158.

**Kata kunci**— Penyakit ginjal kronis, DBSCAN, Epsilon, Cluster

## **Abstract**

Chronic disease (CKD) is a kidney disease characterized by structural or functional kidney damage that lasts more than three months. CKD is characterized by one or more signs of kidney damage, namely albuminuria, abnormal urine sediment, electrolytes, histology, renal structure, or a history of kidney transplantation, with decreased glomerular filtration rate. This study used the DBSCAN implementation method to classify data on diagnosed CKD patients. The essential data used is Chronic Kidney Disease which collects 400 medical data with 24 attributes or features. The study's results note many clusters and noise obtained using Euclidean metrics. The best results are in the second scenario with an epsilon value of 3.5 and  $Min\ sample = 5$ , which produces a total of 2 clusters with a Silhouette value of 0.158.

**Keywords**— Chronic kidney disease, DBSCAN, Epsilon, Cluster



This is an open-access article under the [CC-BY-CA](https://creativecommons.org/licenses/by-sa/4.0/) license

## 1. PENDAHULUAN

Meningkatnya jumlah penderita PGK merupakan masalah kesehatan yang sangat serius. Meningkatnya jumlah pengidap PGK karena bertambahnya jumlah penduduk yang berumur lanjut. Menurut hasil Global Burden of Disease 2010, PGK menempati urutan ke-27 penyebab kematian di dunia pada tahun 1990 dan meningkat menjadi urutan ke-18 pada tahun

2010. Di Indonesia, pengobatan penyakit ginjal merupakan penyedia asuransi kesehatan terbesar kedua setelah penyakit jantung[1] dan 15% orang dewasa yang tinggal di Amerika Serikat menderita PGK [2]. Perawatan primer seperti *screening*, diagnosis, dan manajemen yang tepat oleh dokter diperlukan untuk mencegah PGK yang merugikan, termasuk penyakit kardiovaskular, penyakit ginjal stadium akhir, dan kematian. PGK yang dibiarkan tanpa penanganan yang tepat dapat menyebabkan penyakit ginjal ginjal. PGK harus dicegah dan dikenali dari awal untuk mendapatkan terapi yang efektif [1].

Teknologi *machine learning*(ML) dan *data mining* merupakan bidang penelitian yang dapat mengolah basis data PGK menjadi pengetahuan yang dapat digunakan untuk mendiagnosa PGK. *Data mining* merupakan prosedur untuk menemukan korelasi, pola, dan tren baru dengan menggali data dalam jumlah besar menggunakan statistik dan teknik matematika yang ada. Banyak investasi dari perusahaan besar telah masuk ke dalam *data mining* namun belum menghasilkan banyak keuntungan. Meskipun data berisi beberapa informasi berharga, keberadaannya masih tersembunyi di gudang data [3].

Penelitian-penelitian dibidang ML dan *data mining* untuk diagnosis PGK telah banyak dilakukan. Algoritma Regresi Linier Multivariat dan Regresi Logistik digunakan untuk memastikan seseorang merupakan pasien PGK atau tidak [4]. Penerapan teknik ML (*Naïve Bayes*, *Random Forest*, *Logistic Regression*, dan *Support Vector Machine*) untuk memprediksi gagal ginjal pada tahap awal, basis data penelitian mencakup 316 pasien, proses prediksi mengandalkan spesimen darah dari pasien yang diperoleh dari laboratorium, hasil eksperimen membuktikan bahwa teknik Regresi Logistik telah mencapai akurasi yang tinggi yaitu 97,8% dalam memprediksi penyakit ini [5]. Algoritma seleksi fitur seperti seleksi fitur *Random Forest*, *forward selection*, *forward exhaustive selection*, *backward selection* and *backward exhaustive selection* diidentifikasi dan dievaluasi, kemudian pengklasifikasi ML seperti *Random Forest*, Linear dan *Radial SVM*, *Naïve Bayes* dan *Logistic Regression* diimplementasikan untuk dievaluasi kinerja setiap model ML dalam hal akurasi, sensitivitas, spesifisitas, dan skor AUC. Hasil penelitian menunjukkan bahwa *Random Forest classifier* dengan pemilihan fitur *Random Forest* adalah model pembelajaran mesin yang paling cocok untuk klasifikasi PGK [6].

Penelitian menggunakan teknik *Clustering* (pengelompokan) pada ML untuk diagnosa dan identifikasi PGK juga telah dilakukan. *Clustering* dengan menggunakan algoritma k-mean memanfaatkan *euclidean* serta fungsi jarak manhattan dibandingkan untuk menganalisis data pasien PGK, dan peningkatan jumlah cluster untuk membangun modul prediksi yang baik dalam menentukan pasien PGK dan bukan pasien PGK [7]. Penggunaan teknik seleksi fitur pada basis data PGK dengan algoritma genetik pada tiga teknik *clustering*, yaitu: *K-means*, *Fuzzy C-means* dan *Hierarchical clustering* menghasilkan teknik *clustering* yang terbaik adalah menggunakan *Hierarchical clustering*[8].

Algoritma *Density Based Spatial Clustering Algorithm with Noise (DBSCAN)* digunakan untuk mengelompokkan parameter statiska suhu gabah selama penyimpanan normal sekitar satu tahun dari 27 gudang gabah di China, dimana hasil analisis parameter menggunakan algoritma *DBSCAN* menunjukkan bahwa perbedaan temperatur butir antara lapisan yang berdekatan dan rasio agregasi temperatur butir empat lapis dapat digunakan untuk mendeteksi kekosongan gudang[9]. Algoritma *DBSCAN* untuk *indexing* dan *cosine similarity* untuk proses pencarian cluster yang relevan pada aplikasi *Case-based Reasoning (CBR)* yang telah banyak diterapkan dalam sistem pakar medis. *CBR* memiliki batasan waktu komputasi jika terlalu banyak kasus lama pada basis kasus. Analisis cluster dapat digunakan sebagai algoritma indexing untuk mempercepat pencarian dalam proses pencarian kembali kasus[10]. *DBSCAN* dan rekonstruksi morfologi digunakan untuk mengekstrak segmentasi gabungan untuk meningkatkan segmentasi pembuluh darah retina pada citra medis[11].

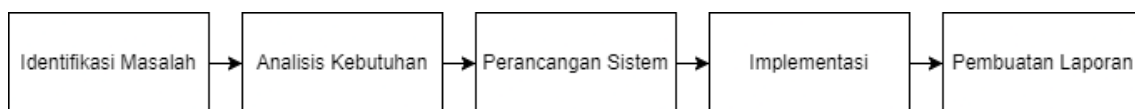
Berdasarkan penelitian-penelitian yang telah dilakukan, maka penelitian ini akan melakukan *clustering* pada data pasien terdiagnosis PGK dengan mengimplementasikan algoritma *clustering DBSCAN* untuk mengelompokkan data pasien terdiagnosis PGK yang memiliki ciri dan karakteristik yang sama dari basis data *Chronic Kidney Disease*.

*Clustering*(pengelompokan) merupakan proses pengelompokkan sejumlah data agar terbentuknya beberapa kelas dengan atribut data masing-masing. Algoritma *density based clustering* adalah algoritma yang paling efisien untuk mendefinisikan *cluster* data dengan kepadatan yang berbeda *clustering*[12].

*DBSCAN* adalah algoritma yang dilandaskan pada kepadatan (*density*) data. *DBSCAN* merupakan contoh pelopor dalam pengembangan teknik *clustering* berbasis kepadatan atau *clustering* berbasis kepadatan[12]. *DBSCAN* tidak perlu mengetahui jumlah kelompok dalam data seperti pada K-Means. Hal ini memberikan keuntungan karena umumnya bentuk dan jumlah kelompok yang sebaiknya diberikan pada data berdimensi tinggi tidak dapat diketahui dengan cara analisis visual data, *DBSCAN* juga dapat mengenali noise dengan baik [13].

## 2. METODE PENELITIAN

Metodologi penelitian yang dilakukan pada penelitian ini seperti yang tersaji pada Gambar 1, terdiri dari beberapa tahapan meliputi: identifikasi masalah, analisis kebutuhan, perancangan sistem, implementasi, dan pelaporan hasil penelitian.



Gambar 1 Bagan Metodologi Penelitian

### 2.1 Identifikasi Masalah

Pada tahap ini, literatur yang terkait dengan topik penelitian ini terlebih dahulu dipelajari dalam bentuk jurnal dan buku yaitu implementasi *clustering* pada PGK menggunakan metode *data mining* dengan algoritma *DBSCAN*. Tujuan dari tahapan ini untuk mengumpulkan literatur sebagai referensi yang relevan, dan dapat memberikan kontribusi terhadap penelitian yang akan dilakukan.

```

RangeIndex: 400 entries, 0 to 399
Data columns (total 24 columns):
 #   Column  Non-Null Count  Dtype
---  -
 0   age     391 non-null    float64
 1   bp      388 non-null    float64
 2   sg      353 non-null    float64
 3   al      354 non-null    float64
 4   su      351 non-null    float64
 5   rbc     248 non-null    object
 6   pc      335 non-null    object
 7   pcc     396 non-null    object
 8   ba      396 non-null    object
 9   bgr     356 non-null    float64
10  bu      381 non-null    float64
11  sc      383 non-null    float64
12  sod     313 non-null    float64
13  pot     312 non-null    float64
14  hemo    348 non-null    float64
15  pcv     330 non-null    object
16  wc      295 non-null    object
17  rc      270 non-null    object
18  htn     398 non-null    object
19  dm      398 non-null    object
20  cad     398 non-null    object
21  appet   399 non-null    object
22  pe      399 non-null    object
23  ane     399 non-null    object
dtypes: float64(11), object(13)
  
```

Gambar 2 Fitur Basis data *Chronic kidney disease*

## 2.2 Analisis Kebutuhan

Pada tahap ini, melakukan pengumpulan data dengan memperoleh data pasien terdiagnosis PGK yang akan dikelompokkan. Basis data yang digunakan pada penelitian ini di ambil dari basis data *Chronic kidney disease*(<https://www.kaggle.com/mansoordaku/ckdisease>). Basis data yang digunakan memiliki 24 atribut atau fitur yang bisa dilihat pada Gambar 2 dengan jumlah data sebanyak 400 data.

## 2.3. Perancangan Sistem

Pada tahap ini dilakukan pembedaan basis data yang dibutuhkan untuk melakukan penelitian. Setelah data dikumpulkan maka akan dilakukan tiga tahapan dari *KDD (Knowledge Discovery in Databases)* pada tahapan ini yaitu

1. *Data Selection* yaitu, pemilihan (seleksi) data dari sekumpulan data yang perlu dilakukan sebelum tahap penggalian informasi dalam *KDD* dan juga data mining. Pada tahap ini peneliti akan menggunakan basis data *Chronic kidney disease*. Selanjutnya akan dilakukan proses *cleaning* dan peneliti akan mengubah semua nama atribut data, karena data yang disediakan masih sulit untuk dimengerti contohnya bp menjadi *blood preassure*.
2. Selanjutnya dilakukan *Pre-processing/Cleaning*, data akan dibersihkan jika ada data yang *null* dan *nan*. Pada tahap ini peneliti akan mengubah data yang mengandung nilai *null* dan *nan* pada *basis data* nilai *null* diubah menjadi nilai 0 agar tidak kehilangan banyak data[14].
3. Setelah itu akan dilakukan *transformation* pada data yang sudah dibersihkan. Data yang telah dipilih akan ditransformasi sehingga data tersebut sesuai untuk proses *data mining*. Proses ini bergantung pada jenis atau pola informasi yang didapatkan dari kedua proses sebelumnya. Mempersiapkan data dari *basis data* yang akan digunakan untuk pengelompokan. Data numerik dari *basis data* akan dianalisis dan fitur-fitur yang terdapat pada basis data akan dilihat mana yang akan dipakai. Pada tahap ini basis data akan di modifikasi dari *basis data* bertipe *object* menjadi *basis data* bertipe *float64*. *Basis data* yang sebelum di ubah menjadi *float* akan di ubah terlebih dahulu isinya karena isinya masih berbentuk *string* sehingga peneliti mengubah data tersebut menjadi numerik, contohnya pada data *red blood cell* memiliki data normal dan abnormal sehingga peneliti akan mengubah normal menjadi angka numerik 1 dan abnormal menjadi angka numerik 2, selanjutnya akan dilakukan normalisasi untuk menyamakan semua nilai *min* dan *max* data sehingga semua data memiliki nilai *min* dan *max* sama yaitu 0 untuk *min* dan 1 untuk *max*. Kemudian akan dilakukan proses *data mining* yang menggunakan algoritma *DBSCAN*.

## 2.4 Implementasi

Setelah data telah disiapkan dan dianalisis, maka pada tahap ini akan dilakukan tahap *KDD* selanjutnya yaitu *data mining*. Data akan dikelompokkan dengan menggunakan algoritma pengelompokan *DBSCAN*. Pengelompokan data menggunakan algoritma *DBSCAN*, dilakukan mengikuti langkah-langkah sebagai berikut:

1. Inisialisasi parameter dan *MinPts* (*MinPts* adalah jumlah minimal pada sample).
2. Menentukan titik parameter.
3. Menghitung semua jarak titik terhadap parameter dan akan menghasilkan *epsilon*.
4. Menentukan titik *Density-reachable* dalam radius berhubungan dengan parameter. Jika jumlah titik *Density-reachable* parameter lebih besar atau sama dengan *MinPts*, maka titik parameter merupakan titik inti, sehingga *cluster* terbentuk dan berlanjut ke titik kerapatan lain disekitar parameter. Jika parameter merupakan titik batas dan tidak ada *Density-reachable* terhadap parameter maka akan dilanjutkan ke titik lain.
5. Ulangi langkah 2 sampai dengan 4 untuk kombinasi parameter dan *MinPts* yang berbeda.

Setelah melakukan tahapan *Data Mining*, hasil *clustering* dari metode masuk ke dalam tahapan terakhir. Tahapan terakhir dari *KDD* yaitu *Interpretation/Evaluation*. Hasil *clustering* pada tahap *data mining* ditampilkan atau divisualisasikan dalam bentuk grafik dan dibandingkan *intercluster* dan *intracluster* masing-masing metode.

*Intercluster* adalah jarak antara dua objek yang masing-masing berada di *cluster* yang berbeda. *Intercluster* dapat dihitung menggunakan persamaan (1), (2), dan (3).

$$Single = \delta_1(S,T) = \min d(x,y) \tag{1}$$

$$Complete = \delta_2(S,T) = \max d(x,y) \tag{2}$$

$$Average = \delta_3(S,T) = \frac{1}{|S||T|} \sum_{x \in S} \sum_{y \in T} d(x,y) \tag{3}$$

Keterangan:

x = objek yang berada dalam *cluster* S

y = objek yang berada dalam *cluster* T

|S| = jumlah objek pada *cluster* S

|T| = jumlah objek pada *cluster* T

d (x, y) = jarak antara x dan y menggunakan distance *metric* yang digunakan.

*Intra cluster* adalah jarak antara dua objek yang berada di *cluster* yang sama. *Intra cluster* dapat dihitung menggunakan persamaan (4).

$$\Delta(S) = d(x,y) \tag{4}$$

Keterangan:

x, y = dua objek yang berada dalam *cluster* yang sama,  $x \neq y$

d (x,y) = jarak antara kedua objek

```

RangeIndex: 400 entries, 0 to 399
Data columns (total 24 columns):
 #   Column                                Non-Null Count  Dtype
---  -
 0   age                                    400 non-null    float64
 1   blood_pressure                        400 non-null    float64
 2   specific_gravity                      400 non-null    float64
 3   albumin                               400 non-null    float64
 4   sugar                                 400 non-null    float64
 5   red_blood_cells                       400 non-null    float64
 6   pus_cell                              400 non-null    float64
 7   pus_cell_clumps                       400 non-null    float64
 8   bacteria                              400 non-null    float64
 9   blood_glucose_random                  400 non-null    float64
10  blood_urea                            400 non-null    float64
11  serum_creatinine                      400 non-null    float64
12  sodium                                 400 non-null    float64
13  potassium                             400 non-null    float64
14  haemoglobin                           400 non-null    float64
15  packed_cell_volume                    400 non-null    float64
16  white_blood_cell_count                 400 non-null    float64
17  red_blood_cell_count                  400 non-null    float64
18  hypertension                          400 non-null    float64
19  diabetes_mellitus                     400 non-null    float64
20  coronary_artery_disease                400 non-null    float64
21  appetite                              400 non-null    float64
22  peda_edema                            400 non-null    float64
23  aanemia                                400 non-null    float64
dtypes: float64(24)
memory usage: 75.1 KB
    
```

Gambar 3 Fitur Basis data *Chronic Kidney Disease* Setelah di ubah

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Akuisisi data

Basis data yang berjumlah 400 pasien dengan 24 atribut atau fitur yang telah dipilih yaitu *age, blood pressure, specific gravity, albumin, sugar, red blood cells, pus cell, pus cell clumps, bacteria, blood glucose random, blood urea, serum creatinine, sodium, potassium, haemoglobin, packed cell volume, white blood cell count, red blood cell count, hypertension, diabetes mellitus, coronary artery disease, appetite, pada edema, anemia* seperti yang tersaji pada pada Gambar 3. Data yang ada pada basis data PGK perlu dilakukan normalisasi, hasil dari normalisasi yang dilakukan disajikan pada Tabel 1. Setiap skenario dilakukan sebanyak 5 kali pengujian dan di catat *cluster* dan *noise* dari hasil pengelompokan setiap skenario dan iterasinya dicatat untuk melakukan evaluasi dan analisis terhadap hasil pengujian pengelompokan yang dilakukan.

Tabel 1. Normalisasi Data yang dilakukan

| No | Data                        | Sebelum di Normalisasi |            | Sesudah di Normalisasi |            |
|----|-----------------------------|------------------------|------------|------------------------|------------|
|    |                             | <i>min</i>             | <i>max</i> | <i>min</i>             | <i>max</i> |
| 1  | <i>age</i>                  | 2                      | 90         | 0                      | 1          |
| 2  | <i>blood_pressure</i>       | 50                     | 180        | 0                      | 1          |
| 3  | <i>specific_gravity</i>     | 1,005                  | 1,025      | 0                      | 1          |
| 4  | <i>albumin</i>              | 0                      | 5          | 0                      | 1          |
| 5  | <i>sugar</i>                | 0                      | 5          | 0                      | 1          |
| 6  | <i>blood_glucose_random</i> | 22                     | 490        | 0                      | 1          |
| 7  | <i>blood_urea</i>           | 1.5                    | 391        | 0                      | 1          |
| 8  | <i>serum_creatinine</i>     | 0,4                    | 76         | 0                      | 1          |
| 9  | <i>sodium</i>               | 4,5                    | 163        | 0                      | 1          |
| 10 | <i>potassium</i>            | 2,5                    | 47         | 0                      | 1          |

#### 3.2 Hasil Pengujian

Berdasarkan pengujian yang telah dilakukan, hasil yang didapat dari pengelompokan data pasien terdiagnosis PGK menggunakan algoritma *DBSCAN*. Pengujian dilakukan dengan menentukan jumlah *cluster* menggunakan metric *Euclidean*. Parameter yang digunakan dalam 5 skenario yang dilakukan dapat dilihat pada Tabel 2

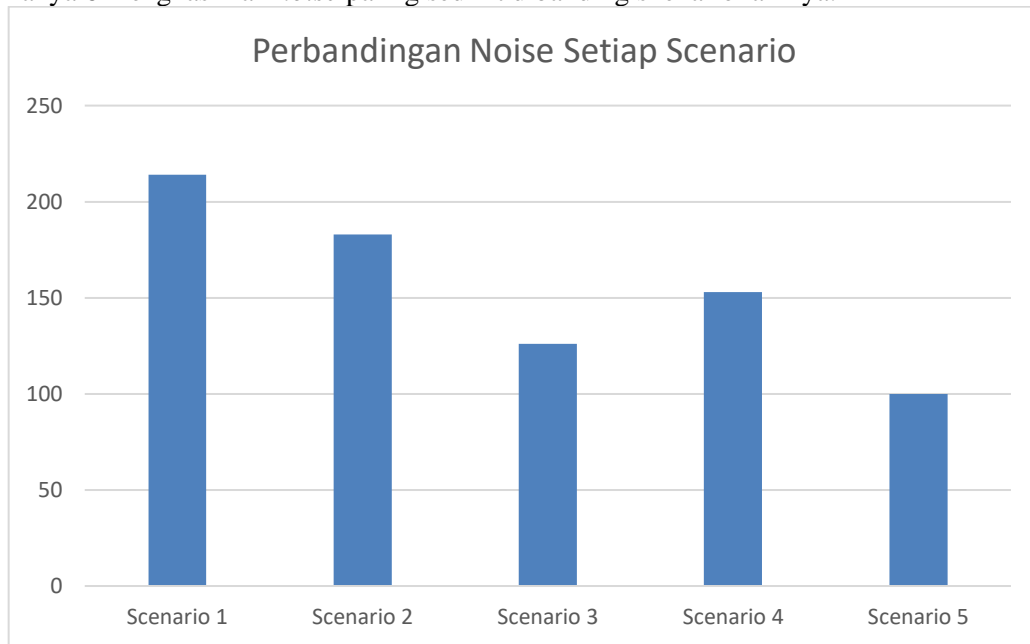
Tabel 2. Parameter 5 Skenario yang dilakukan

| skenario | <i>epsilon</i> | <i>Min sample</i> | Jumlah <i>cluster</i> |
|----------|----------------|-------------------|-----------------------|
| 1        | 3              | 5                 | 3                     |
| 2        | 3,5            | 5                 | 2                     |
| 3        | 4              | 5                 | 3                     |
| 4        | 3,5            | 3                 | 8                     |
| 5        | 4              | 3                 | 6                     |

Skenario pertama menggunakan parameter *epsilon* 3 dengan *min sample* 5 dan menghasilkan *cluster* sebanyak 3 *cluster* seperti ditunjukkan pada tabel 2. Untuk skenario kedua menggunakan parameter *epsilon* 3,5 dengan *min sample* 5 dan menghasilkan *cluster* sebanyak 2 *cluster*. Untuk percobaan tiga menggunakan parameter *epsilon* 4 dengan *min sample* 5 dan menghasilkan *cluster* sebanyak 3 *cluster*. Untuk skenario ke empat menggunakan parameter *epsilon* 3,5 dengan *min sample* 3 dan menghasilkan *cluster* sebanyak 8 *cluster* untuk skenario

kelima menggunakan parameter *epsilon* 4 dengan *min sample* 3 dan menghasilkan *cluster* sebanyak 6 *cluster*.

Hasil dari setiap skenario yang meliputi skenario pertama sampai skenario kelima dalam proses *clustering* menggunakan *basis data Chronic Kidney Disease* dapat menghasilkan nilai *noise*. Nilai *noise* tersebut dibandingkan untuk melihat skenario ke berapa yang memiliki *noise* atau data yang tidak termasuk dari *cluster*. Pada gambar 4 menunjukkan jika skenario 5 dengan *cluster* hanya 6 menghasilkan *noise* paling sedikit dibanding skenario lainnya.



Gambar 4 Grafik Perbandingan *Noise* setiap Skenario

### 3.3 Analisa Hasil Pengujian

Pada pengujian menggunakan 5 skenario dalam proses *clustering* menggunakan basis data *Chronic Kidney Disease* dapat dikatakan jika jumlah *cluster* tidak mempengaruhi banyaknya nilai *noise* yang dihasilkan dan untuk visualisasi pada *DBSCAN* menggunakan *Scatterplot* menghasilkan hasil yang kurang efektif untuk melihat data *cluster* karena data yang digunakan pada penelitian ini termasuk besar karena memiliki 24 atribut data dan masing-masing atribut memiliki 400 data. Untuk hasil *Silhouette Coefficient* bisa dilihat pada Gambar 4 yang menunjukkan bahwa banyaknya nilai *cluster* tidak selalu mempengaruhi nilai *Silhouette* dari hasil *clustering* yang dilakukan.

Hasil *Silhouette* yang didapat dari Tabel 3 dapat menunjukkan bahwa skenario yang mendapatkan nilai *silhouette* terbaik adalah skenario ke 2 yang memiliki nilai *silhouette* terbesar yaitu 0.15801077304214137 dengan nilai *epsilon* 3,5 dan *minSample* sebesar 5 untuk mendapatkan *cluster* sebanyak 2 *cluster*.

Tabel 3 Hasil *Silhouette Coefficient* setiap Skenario

| No | Skenario   | Hasil <i>Silhouette</i> |
|----|------------|-------------------------|
| 1  | Skenario 1 | 0.0529284816119996      |
| 2  | Skenario 2 | 0.1580107730421413      |
| 3  | Skenario 3 | 0.1376267256736861      |
| 4  | Skenario 4 | 0.0967843607689665      |
| 5  | Skenario 5 | 0.0971374370147013      |

Hasil *InterCluster* yang didapat pada Tabel 4 dapat menunjukkan bahwa skenario yang mendapatkan nilai terbaik adalah skenario pertama yang memiliki nilai rata-rata terkecil yaitu 1.880045 yang menggunakan *epsilon* sebesar 3 dan *minSample* sebesar 5 untuk mendapatkan cluster sebanyak 3 cluster.

Tabel 4 Hasil *inter cluster* setiap Skenario

| No | Nilai                         | Skenario1 | Skenario2 | Skenario3 | Skenario4 | Skenario5 |
|----|-------------------------------|-----------|-----------|-----------|-----------|-----------|
| 1  | <i>Inter Cluster</i> Terkecil | 0.301114  | 0.407001  | 0.525654  | 0.407001  | 1.135819  |
| 2  | <i>Inter Cluster</i> Terbesar | 5.379030  | 5.475547  | 5.574571  | 6.552946  | 6.552946  |
|    | Rata-rata                     | 1.880045  | 2.000273  | 2.258307  | 2.126972  | 2.531345  |

Tabel 5 Hasil *intra cluster* setiap Skenario

| No | Nilai                         | Skenario1 | Skenario2 | Skenario3 | Skenario4 | Skenario5 |
|----|-------------------------------|-----------|-----------|-----------|-----------|-----------|
| 1  | <i>Intra Cluster</i> Terkecil | 0.307337  | 0.322999  | 0.883223  | 0.469104  | 0.786526  |
| 2  | <i>Intra Cluster</i> Terbesar | 1.002936  | 0.927748  | 1.444650  | 0.940481  | 1.862444  |
|    | Rata-rata                     | 0.654638  | 0.703991  | 1.123702  | 0.744405  | 1.277502  |

Hasil *IntraCluster* yang didapat pada tabel 5 dapat menunjukkan bahwa skenario yang mendapatkan nilai terbaik adalah skenario kelima yang memiliki rata-rata terbesar yang mendapatkan hasil sebesar 1.277502 yang menggunakan *epsilon* sebesar 4 dan *minSample* sebesar 3 untuk mendapatkan cluster sebanyak 6 cluster. Hasil *clustering* yang baik adalah yang memiliki *intra cluster distance* yang besar dan *inter cluster* yang kecil, sehingga diperoleh *inter cluster* terkecil adalah 0,301114 dan *intra cluster* terbesar adalah 1,862444.

#### 4. KESIMPULAN

Hasil penelitian menunjukkan bahwa implementasi algoritma DBSCAN pada basis data pasien terdiagnosa PGK nilai *epsilon* dan nilai *MinPts* sangat berpengaruh terhadap hasil clustering untuk basis data pasien terdiagnosa PGK, penggunaan *ScatterPlot* pada algoritma DBSCAN untuk basis data pasien terdiagnosa PGK menghasilkan jumlah yang banyak sehingga sulit untuk melihat semua isi *cluster*. Pada penelitian ini didapatkan nilai *Silhouette* terbesar pada skenario kedua, yaitu skenario yang menghasilkan 2 *cluster* dengan nilai *Epsilon* = 3,5 dan *MinSample* = 5, dan didapatkan nilai *intercluster* terbaik yang didapatkan oleh skenario pertama yang mendapatkan hasil 3 *cluster* dengan nilai *Epsilon* = 3 dan *MinSample* = 5 sedangkan untuk *intracluster* terbaik didapatkan oleh skenario kelima yang mendapatkan hasil 6 *cluster* dengan nilai *Epsilon* = 4 dan *MinSample* = 3.

#### DAFTAR PUSTAKA

- [1] V. K. Gliselda, "Diagnosis dan Manajemen Penyakit Ginjal Kronis (PGK)," *J. Med. Hutama*, 2021, [Daring]. Tersedia pada: <http://www.jurnalmedikahutama.com/index.php/JMH/article/view/237>.
- [2] C. for D. C. and Prevention, "Chronic kidney disease in the United States, 2019," *Fluoride Action Netw.*, 2019, [Daring]. Tersedia pada: <https://fluoridealert.org/studytracker/38332/>.
- [3] S. Sugiono, S. Nurdiani, S. Linawati, dan ..., "Pengelompokan Perilaku Mahasiswa Pada Perkuliahan E-Learning dengan K-Means Clustering," *J. Kaji. ...*, 2019, [Daring].



- Tersedia pada: <http://jurnal.ubharajaya.ac.id/index.php/kajian-ilmiah/article/view/410>.
- [4] B. Singh, K. Arora, dan S. S. Iyer, “Chronic Kidney Disease Detection Using Machine Learning Regression Models,” *ECS Trans.*, vol. 107, no. 1, hal. 2191, 2022, doi: 10.1149/10701.2191ecst.
- [5] M. M. Mijwil dan A. S. Alsaadi, “Applying Machine Learning Techniques to Predict Chronic Kidney Disease in Humans,” 2021, [Daring]. Tersedia pada: [https://www.researchgate.net/profile/Maad-Mijwil/publication/354380202\\_Applying\\_Machine\\_Learning\\_Techniques\\_to\\_Predict\\_Chronic\\_Kidney\\_Disease\\_in\\_Humans/links/6135c0942b40ec7d8bead4ab/Applying-Machine-Learning-Techniques-to-Predict-Chronic-Kidney-Disease-i](https://www.researchgate.net/profile/Maad-Mijwil/publication/354380202_Applying_Machine_Learning_Techniques_to_Predict_Chronic_Kidney_Disease_in_Humans/links/6135c0942b40ec7d8bead4ab/Applying-Machine-Learning-Techniques-to-Predict-Chronic-Kidney-Disease-i).
- [6] A. A. Abdullah, S. A. Hafidz, dan W. Khairunizam, “Performance Comparison of Machine Learning Algorithms for Classification of Chronic Kidney Disease (CKD),” *J. Phys. Conf. Ser.*, vol. 1529, no. 5, hal. 52077, 2020, doi: 10.1088/1742-6596/1529/5/052077.
- [7] R. S. Walse, G. D. Kurundkar, P. U. Bhalchandra, dan S. A. S. Alshaibani, “Prediction of Chronic Kidney Disease Data Using Clustering Techniques—A Comparative Approach BT - Proceeding of First Doctoral Symposium on Natural Computing Research,” in *Proceeding of First Doctoral Symposium on Natural Computing Research*, 2021, hal. 451–461.
- [8] S. K. Sahu dan A. K. Shrivastava, “Analysis and Comparison of Clustering Techniques for Chronic Kidney Disease With Genetic Algorithm,” *Int. J. Comput. Vis. ...*, 2018, [Daring]. Tersedia pada: <https://www.igi-global.com/article/analysis-and-comparison-of-clustering-techniques-for-chronic-kidney-disease-with-genetic-algorithm/214071>.
- [9] H. Cui, W. Wu, Z. Zhang, F. Han, dan Z. Liu, “Clustering and application of grain temperature statistical parameters based on the DBSCAN algorithm,” *J. Stored Prod. Res.*, vol. 93, hal. 101819, 2021, doi: <https://doi.org/10.1016/j.jspr.2021.101819>.
- [10] H. Santoso dan A. Musdholifah, “Case Base Reasoning (CBR) and Density Based Spatial Clustering Application with Noise (DBSCAN)-based Indexing in Medical Expert Systems,” *Khazanah Inform. J. Ilmu ...*, 2019, [Daring]. Tersedia pada: <https://journals.ums.ac.id/index.php/khif/article/view/8323>.
- [11] K. Mardani dan K. Maghooli, “Enhancing retinal blood vessel segmentation in medical images using combined segmentation modes extracted by DBSCAN and morphological reconstruction,” *Biomed. Signal Process. Control*, vol. 69, hal. 102837, 2021, doi: <https://doi.org/10.1016/j.bspc.2021.102837>.
- [12] D. Safitri, T. Wuryandari, dan ..., “Metode DBSCAN Untuk Pengelompokan Kabupaten/Kota di Provinsi Jawa Tengah Berdasarkan Produksi Padi Sawah dan Padi Ladang,” *J. Stat. ...*, 2017, [Daring]. Tersedia pada: <https://jurnal.unimus.ac.id/index.php/statistik/article/view/2710>.
- [13] F. Nur, *Analisis Kinerja Metode Dbscan (Density-Based Spatial Clustering of Applications with Noise) dan K-Means dalam Sistem Pendukung Keputusan*. repositori.usu.ac.id, 2017.
- [14] P. Gupta, *Practical Data Science with Jupyter: Explore Data Cleaning, Pre-processing, Data Wrangling, Feature Engineering and Machine Learning using Python and Jupyter (English Edition)*. BPB PUBN, 2021.